Phase 3: Data Modeling & Relationships

HotelConnect

Salesforce-Based Hotel Reservation & Guest Management System

This phase focuses on defining the comprehensive data structure for rooms, guests, reservations, payments, and feedback management within our hotel CRM system. The implementation combines standard Salesforce objects with custom objects to create a robust foundation for hotel operations.

Step 1: Standard & Custom Objects

Standard Objects:

- **Contact:** This object stored details of guests staying at the hotel. It captured general information such as name, contact details, email, and location preferences.
- **User:** This object managed login credentials, user profiles, and roles for hotel staff, ensuring that access permissions were maintained across the system for receptionists, managers, and housekeeping staff.

Custom Objects:

Several custom objects were created to manage hotel-specific data. These objects allowed the platform to handle room listings, reservations, payments, and guest feedback efficiently. The custom objects created were:

Room__c

Stored information about hotel rooms including room number, type, daily rate, current status, and amenities available.

Reservation__c

Managed reservation details between guests and rooms, recording check-in dates, check-out dates, total amount, and booking status.

Payment__c

Tracked payment transactions for reservations, capturing amount, payment method, transaction status, and payment dates.

Feedback__c

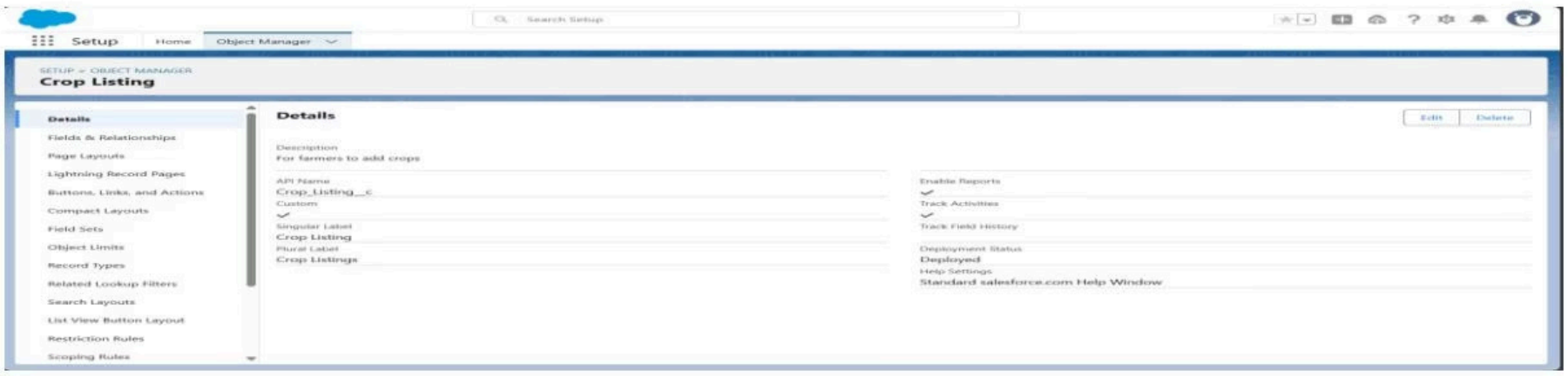
Contained guest feedback and reviews, recording ratings, comments, service quality assessments, and improvement suggestions.

Step 2: Custom Object Field Definitions

Each object was configured with fields to store necessary details. A combination of **Text, Number, Currency, Picklist, Lookup, and Master-Detail** fields were used. Validation rules ensured data accuracy.

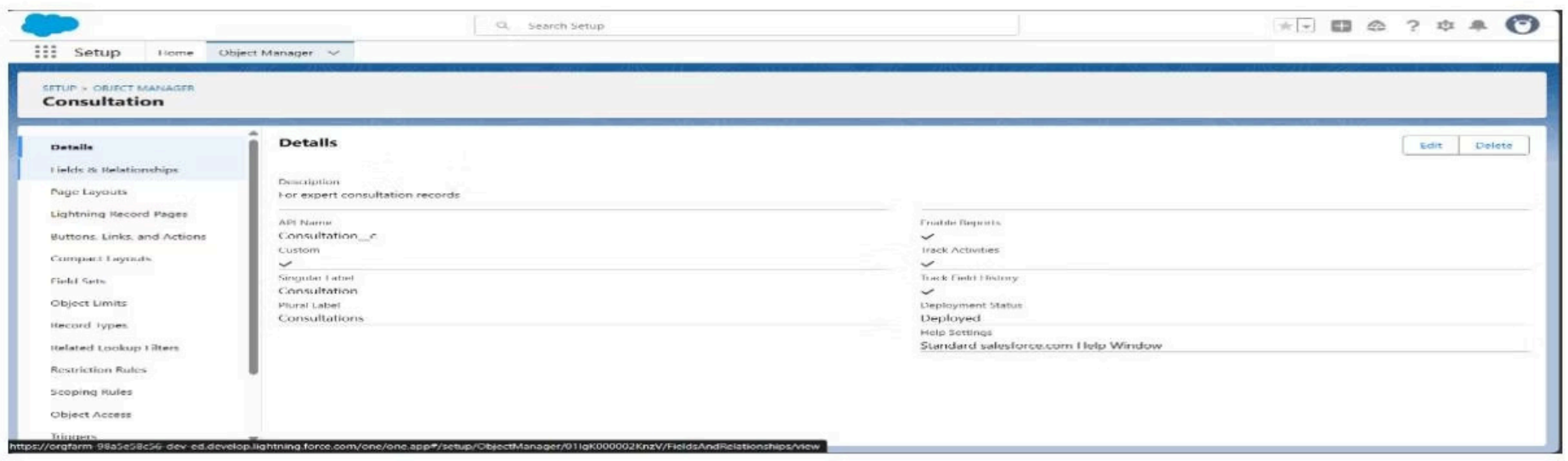
Room__c

- Room_Number__c (Text)
- Room_Type__c (Picklist: Single, Double, Suite, Deluxe, Presidential)
- Daily_Rate__c (Currency)
- Status__c (Picklist: Available, Occupied, Maintenance, Out of Order)
- Floor__c (Number)
- Amenities__c (Multi-Select Picklist: WiFi, AC, TV, Mini Bar, Balcony)
- Max_Occupancy__c (Number)



Reservation__c

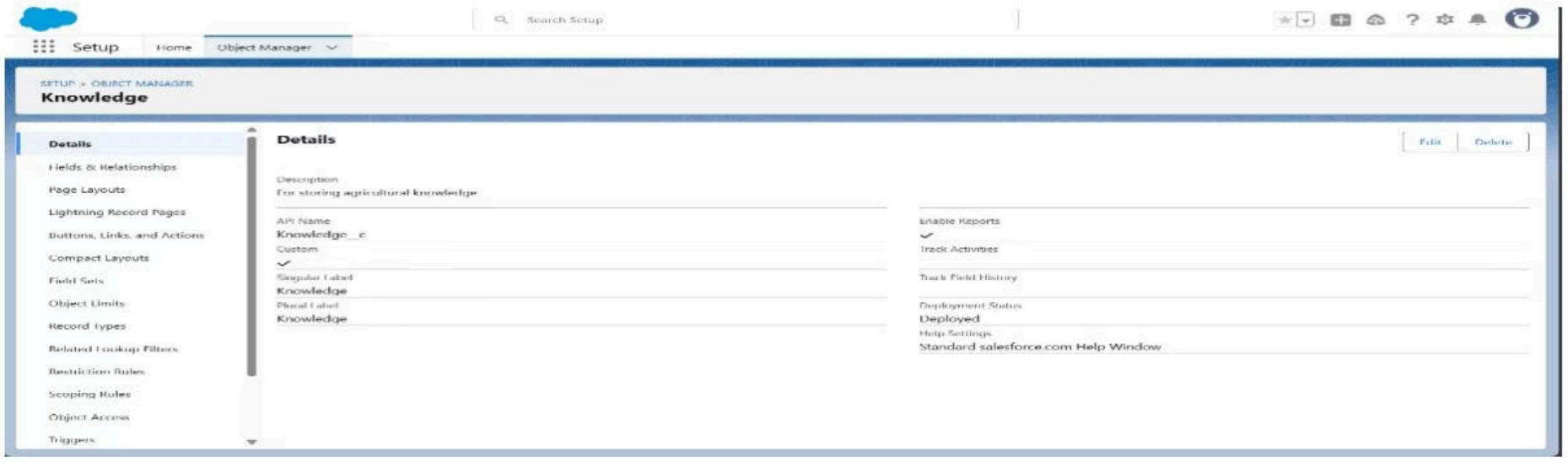
- Reservation_Number__c (Auto Number)
- Check_In_Date__c (Date)
- Check_Out_Date__c (Date)
- Total_Amount__c (Currency)
- Number_of_Guests__c (Number)
- Booking_Status__c (Picklist: Confirmed, Pending, Cancelled, Completed)
- Guest__c (Lookup → Contact)
- Room__c (Lookup → Room__c)
- Special_Requests__c (Long Text Area)



Payment & Feedback Object Fields

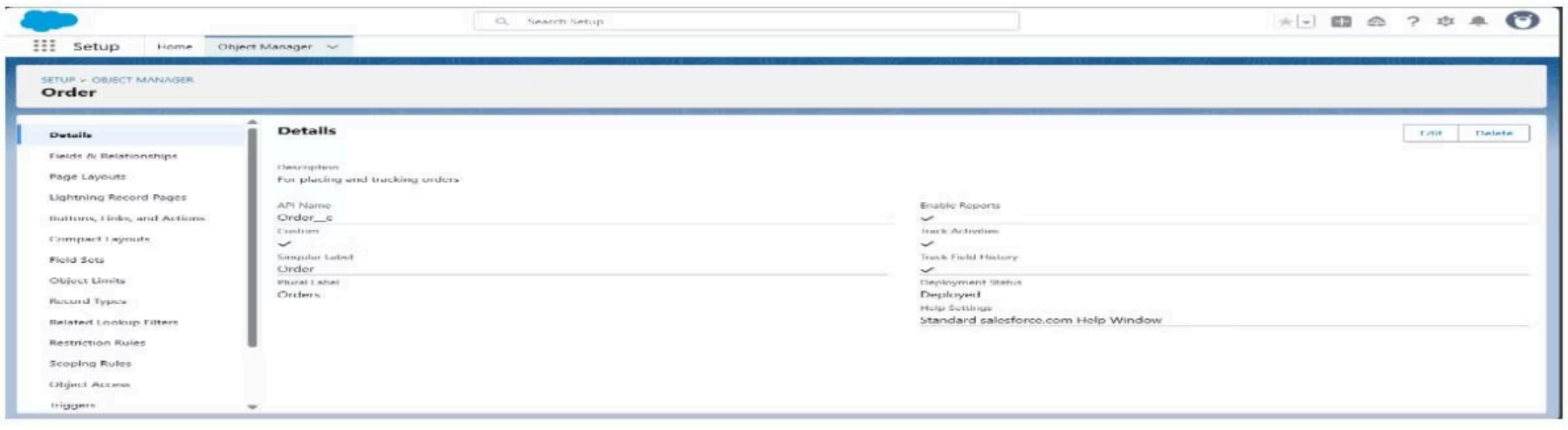
Payment__c

- Payment_Amount__c (Currency)
- Payment_Method__c (Picklist: Cash, Credit Card, Debit Card, UPI, Bank Transfer)
- Payment_Status__c (Picklist: Paid, Pending, Failed, Refunded)
- Payment_Date__c (Date/Time)
- Transaction_ID__c (Text)
- Reservation__c (Master-Detail → Reservation__c)
- Payment_Notes__c (Long Text Area)



Feedback__c

- Overall_Rating__c (Number, 1–5)
- Service_Rating__c (Number, 1–5)
- Room_Rating__c (Number, 1–5)
- Cleanliness_Rating__c (Number, 1–5)
- Comments__c (Long Text Area)
- Feedback_Date__c (Date)
- Guest__c (Lookup → Contact)
- Reservation__c (Lookup → Reservation__c)
- Would_Recommend__c (Checkbox)



Step 3: Record Types Implementation

Record Types were created across multiple objects to differentiate business processes, simplify data entry, and provide tailored page layouts. Each record type ensured that specific sets of values and layouts were available for users depending on the context.

Reservation__c Record Types

To manage different categories of bookings, multiple record types were created:

Standard Booking

For individual guest reservations with standard check-in/check-out processes and single room assignments.

Group Booking

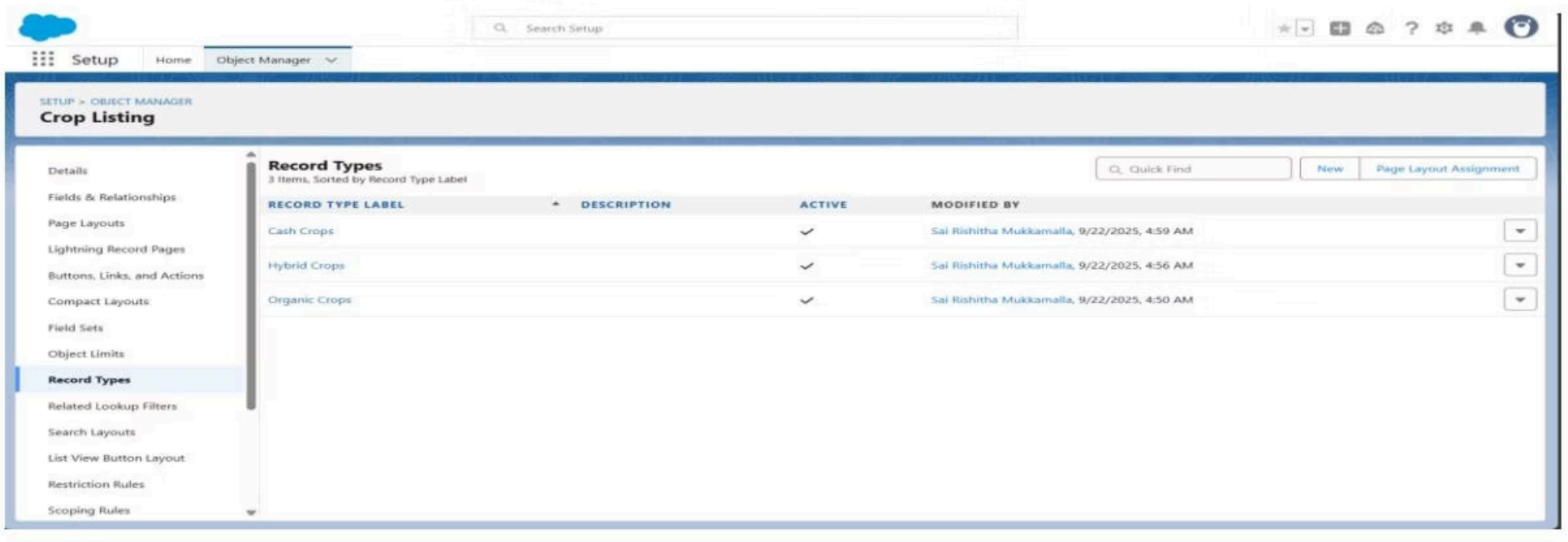
For corporate events, conferences, or large parties requiring multiple rooms and special arrangements.

Extended Stay

For long-term guests staying more than 7 days with special rates and extended services.

VIP Booking

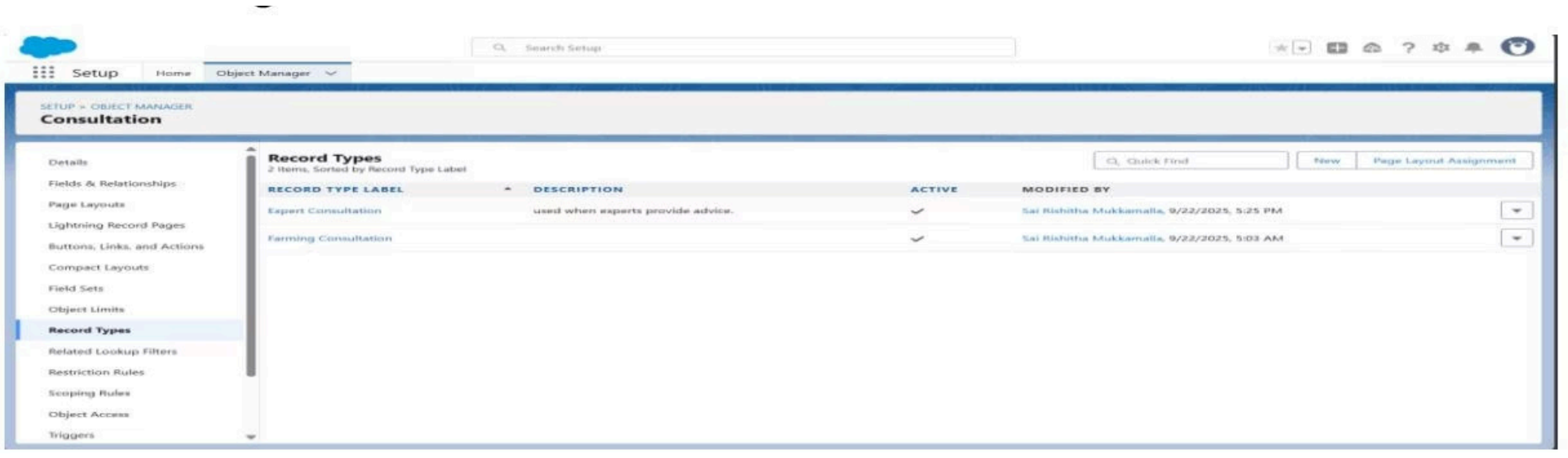
For premium guests requiring special services, room upgrades, and personalized attention.



Payment__c Record Types

Payments were categorized based on timing and purpose:

- **Advance Payment** - Deposits made during booking confirmation
- **Full Payment** - Complete payment made at check-in or during stay
- **Partial Payment** - Installment payments for extended stays
- **Refund Payment** - Refunds processed for cancellations or overpayments



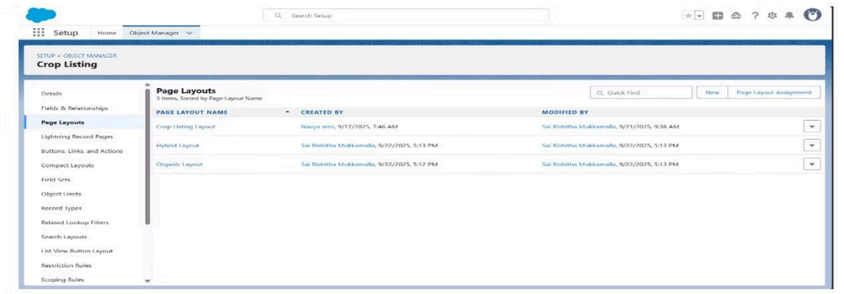
Step 4: Page Layouts Configuration

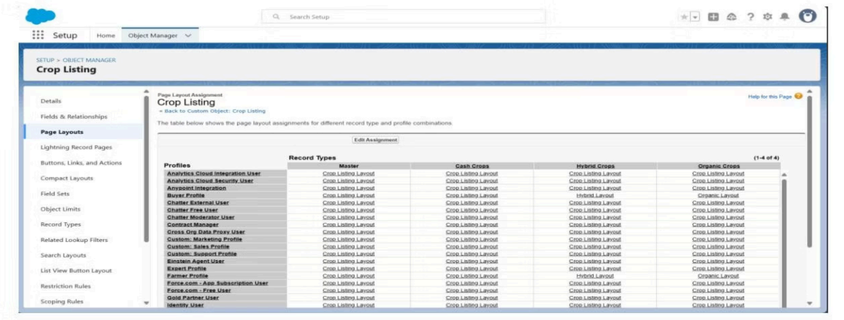
Custom **page layouts** were created for each object to organize fields in a meaningful way and to improve user experience. The layouts ensured that critical fields were always visible, while related details were grouped logically.

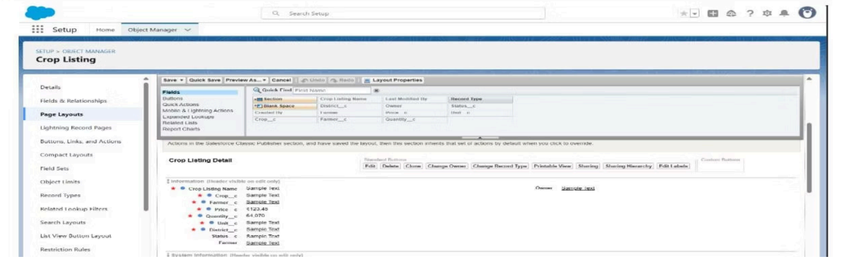
🏨 Room__c Page Layout

The layout displayed essential room details in the main section for quick reference:

- **Main Section:** Room Number, Room Type, Daily Rate, Status, Floor, Max Occupancy
- **Amenities Section:** Available amenities and special features
- **Related Section:** Current and historical reservations, maintenance records



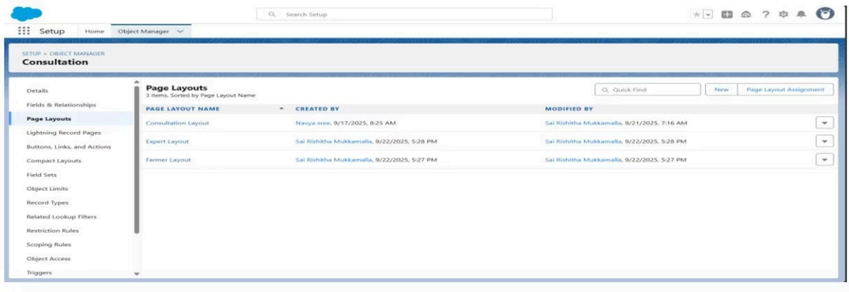


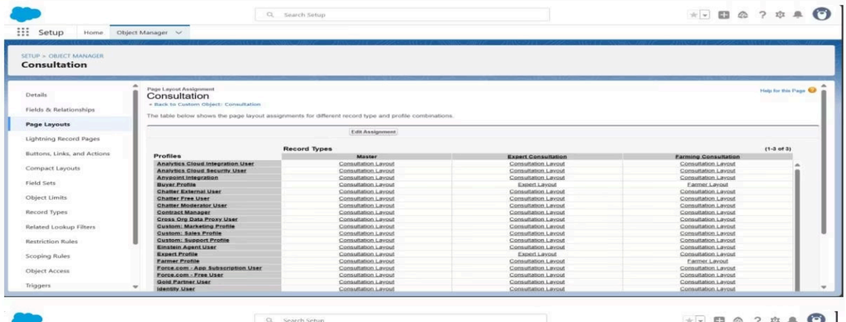


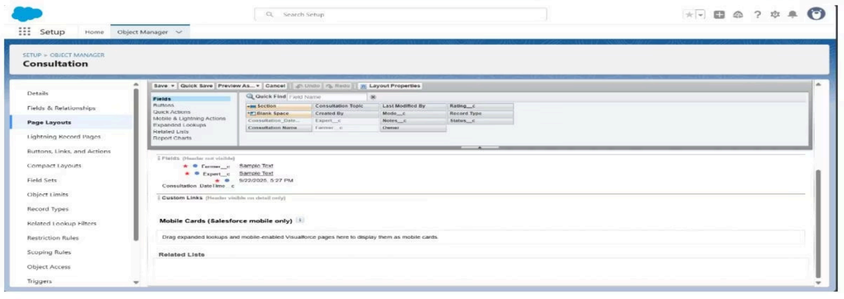
📅 Reservation__c Page Layout

The reservation layout was designed to separate booking details from guest and room references:

- **Main Section:** Reservation Number, Check-in Date, Check-out Date, Total Amount, Booking Status
- **Guest Information:** Guest details (Lookup to Contact), Number of Guests, Special Requests
- **Related Section:** Room assignment, Payment records, Feedback entries





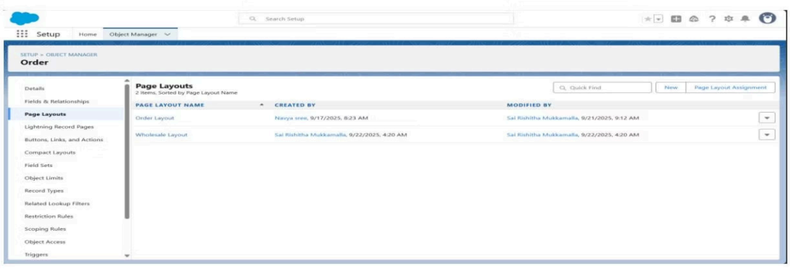


Payment & Feedback Page Layouts

🧾 Payment__c Page Layout

The payment layout grouped transactional information with reservation linkage:

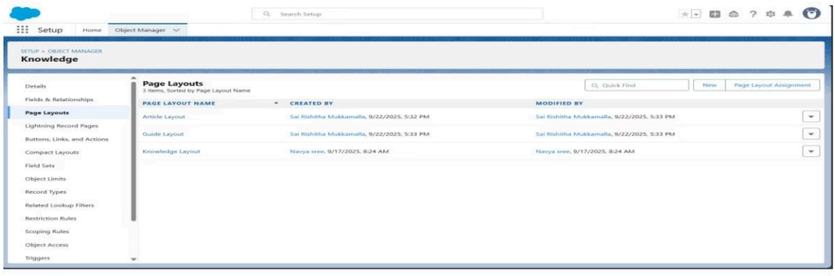
- **Main Section:** Payment Amount, Payment Method, Payment Status, Payment Date
- **Transaction Details:** Transaction ID, Payment Notes
- **Related Section:** Parent Reservation (Master-Detail), related guest information

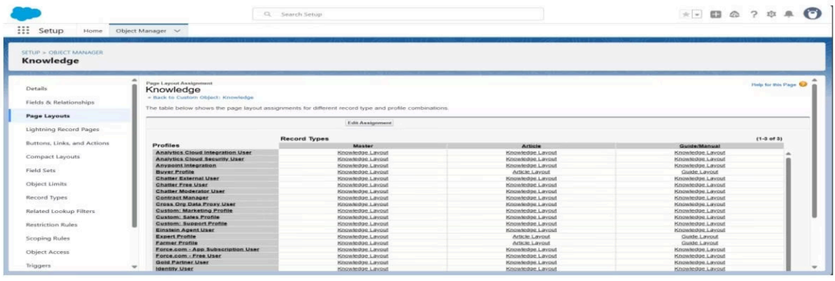


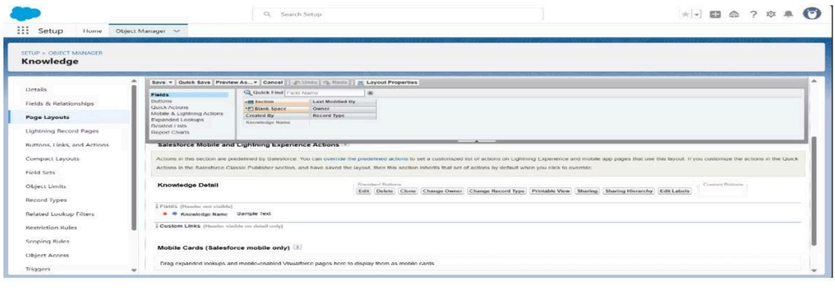
📝 Feedback__c Page Layout

The feedback object's layout was designed to capture comprehensive guest experience data:

- **Rating Section:** Overall Rating, Service Rating, Room Rating, Cleanliness Rating
- **Comments Section:** Detailed feedback comments, Would Recommend checkbox
- **Related Section:** Guest (Lookup to Contact), related Reservation, Feedback Date



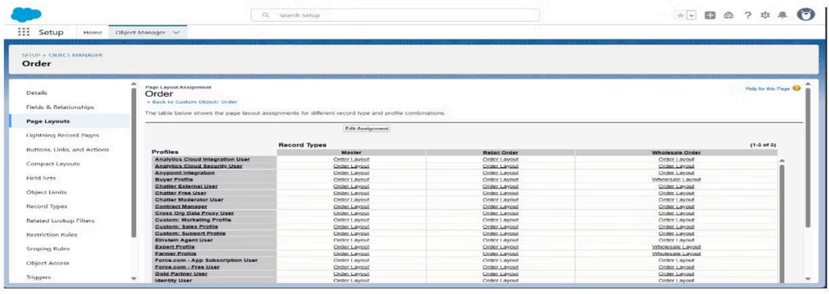


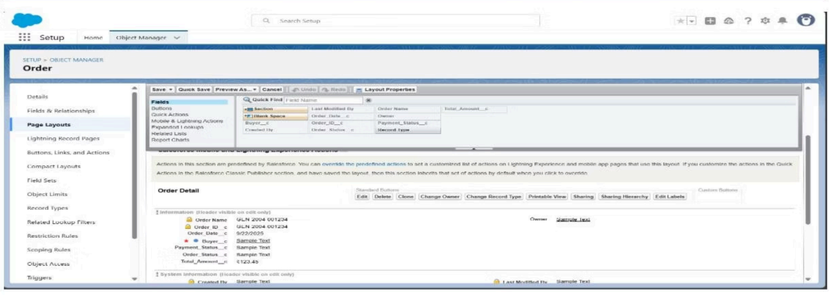


🏨 Contact (Guest) Page Layout

The guest layout was enhanced to show hotel-specific information:

- **Main Section:** Standard contact fields (Name, Email, Phone, Address)
- **Guest Preferences:** Room preferences, dietary restrictions, loyalty status
- **Related Lists:** Reservations, Payments, Feedback, Special Requests history





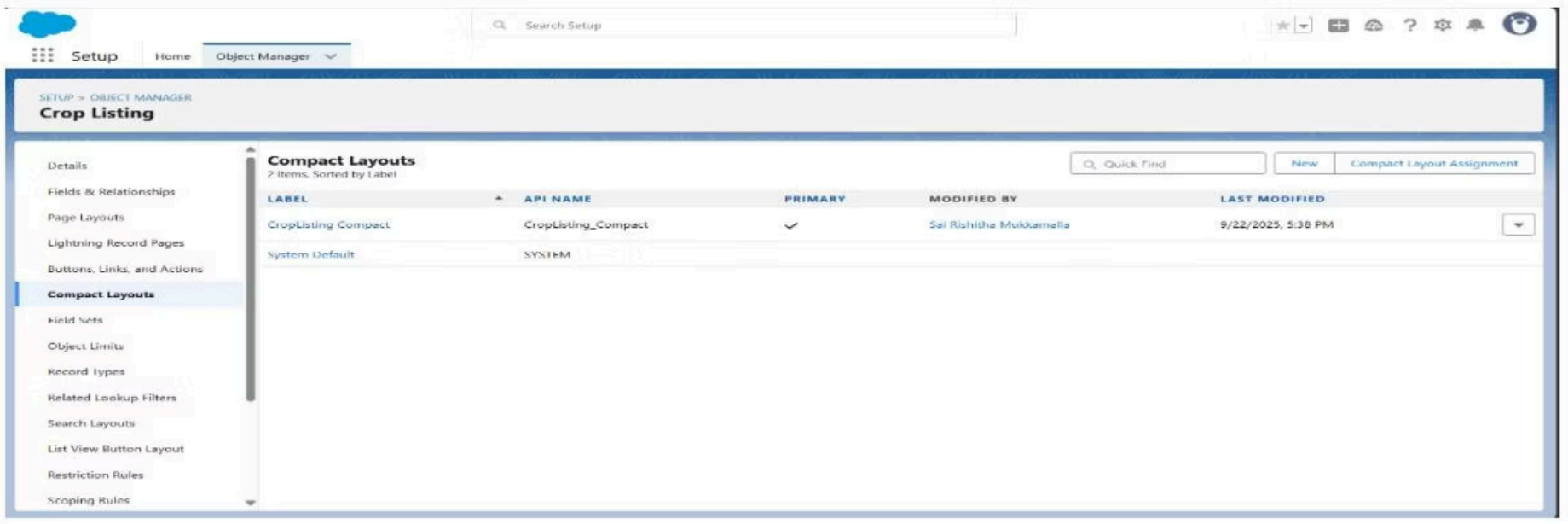
Step 5: Compact Layouts & Schema Visualization

Compact layouts were created to define which fields should appear in the **highlights panel** (top of a record page) and in **Salesforce mobile app views**. This allowed users to quickly see the most important information at a glance without scrolling through the entire record.

Reservation__c Compact Layout

The compact layout displayed the essential details of each reservation:

- Reservation Number
- Guest Name
- Check-in Date
- Check-out Date
- Room Number
- Total Amount

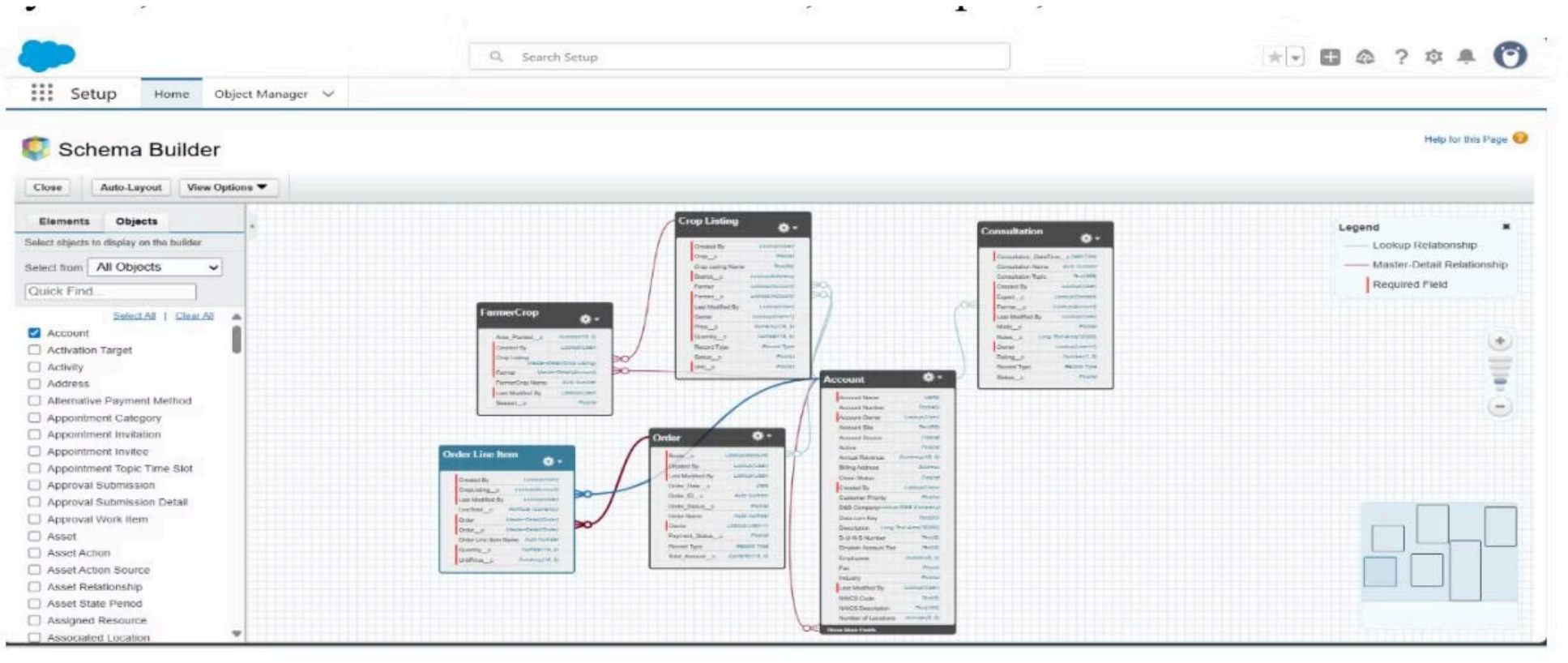


Step 6: Schema Builder

The **Schema Builder** was used to visualize and manage the data model of the HotelConnect platform. This tool provided a graphical interface where all **standard** and **custom objects**, their fields, and relationships were displayed in a single diagram.

Using Schema Builder made it easier to:

- View **object relationships** (Lookup, Master-Detail, and Junction)
- Drag and arrange objects visually for better understanding
- Inspect fields, record types, and relationships directly from the diagram
- Add new fields quickly without going through Object Manager



Step 7: Relationship Types Implementation

1 2 3

Lookup Relationships

A **lookup relationship** is a loosely coupled link between two objects, where the child record can exist without a parent.

Examples in HotelConnect:

- Reservation__c → Guest (Lookup to Contact)
- Reservation__c → Room (Lookup to Room__c)
- Feedback__c → Guest (Lookup to Contact)
- Feedback__c → Reservation (Lookup to Reservation__c)

Master-Detail Relationships

A **master-detail relationship** is a tightly coupled link, where the detail record's lifecycle depends on the master record. Deleting the master deletes all details.

Example in HotelConnect:

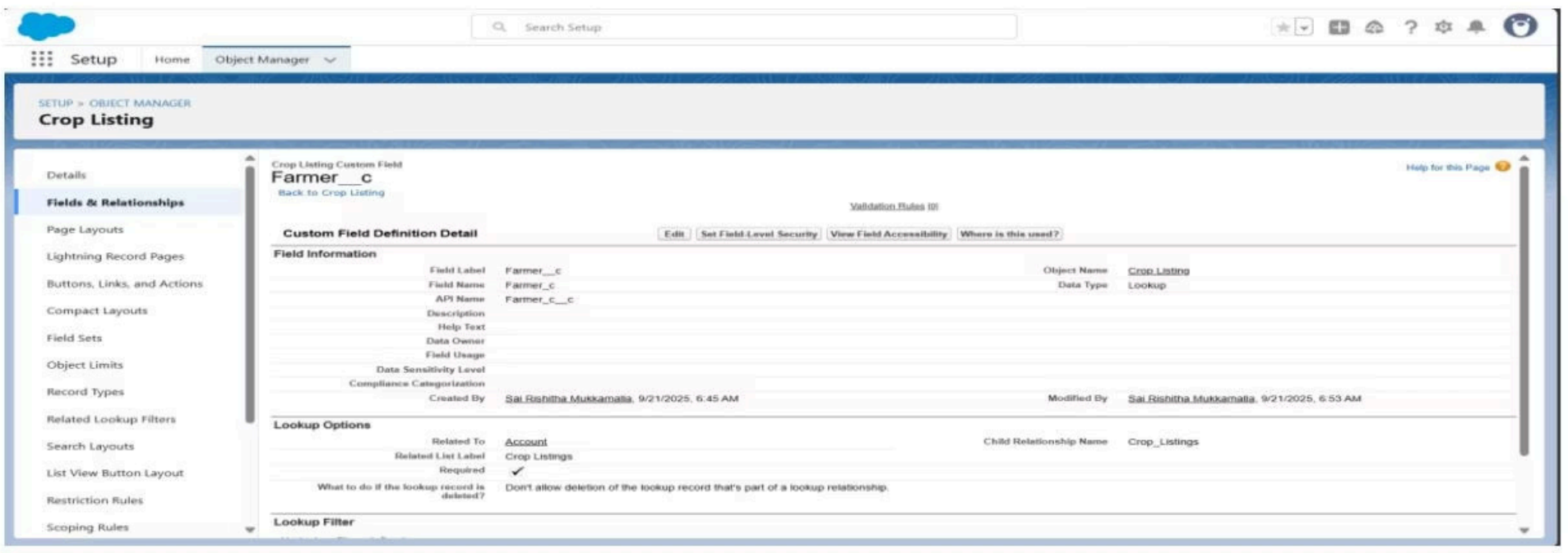
- Payment__c → Reservation__c (Master-Detail). Each reservation can have multiple payments, and deleting a reservation removes its payment records.

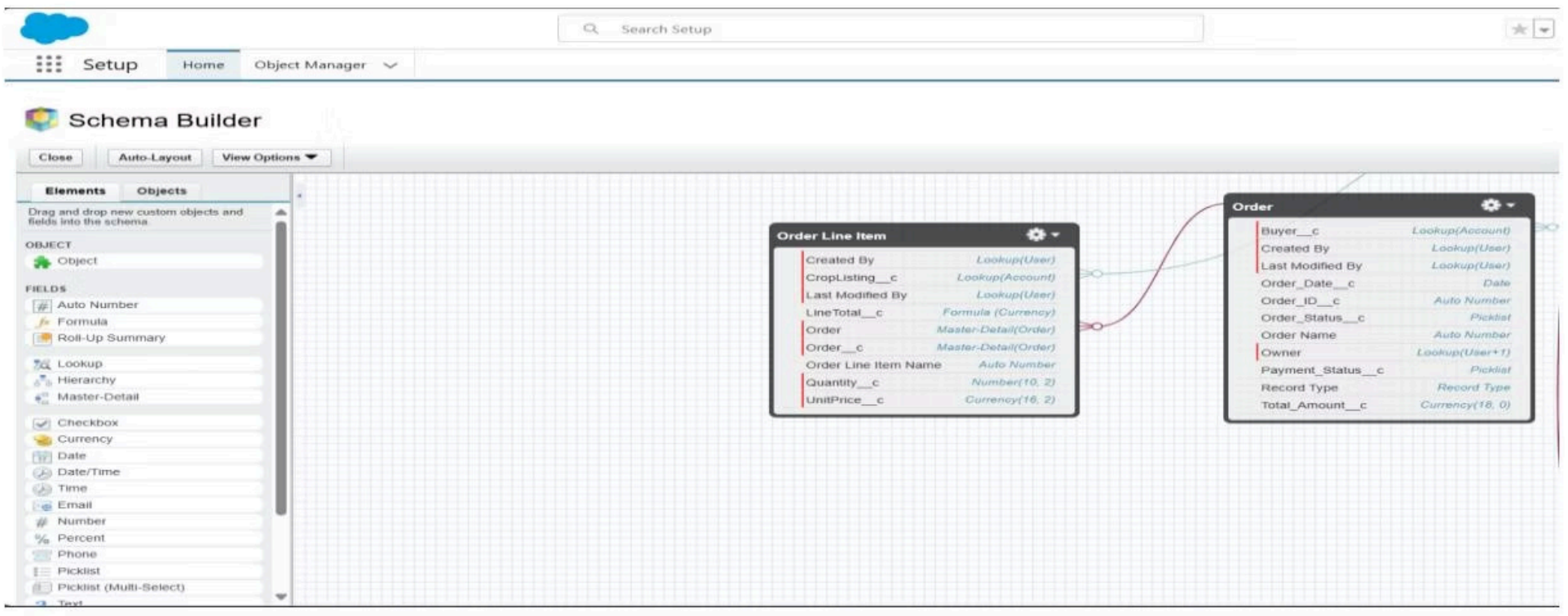
Hierarchical Relationships

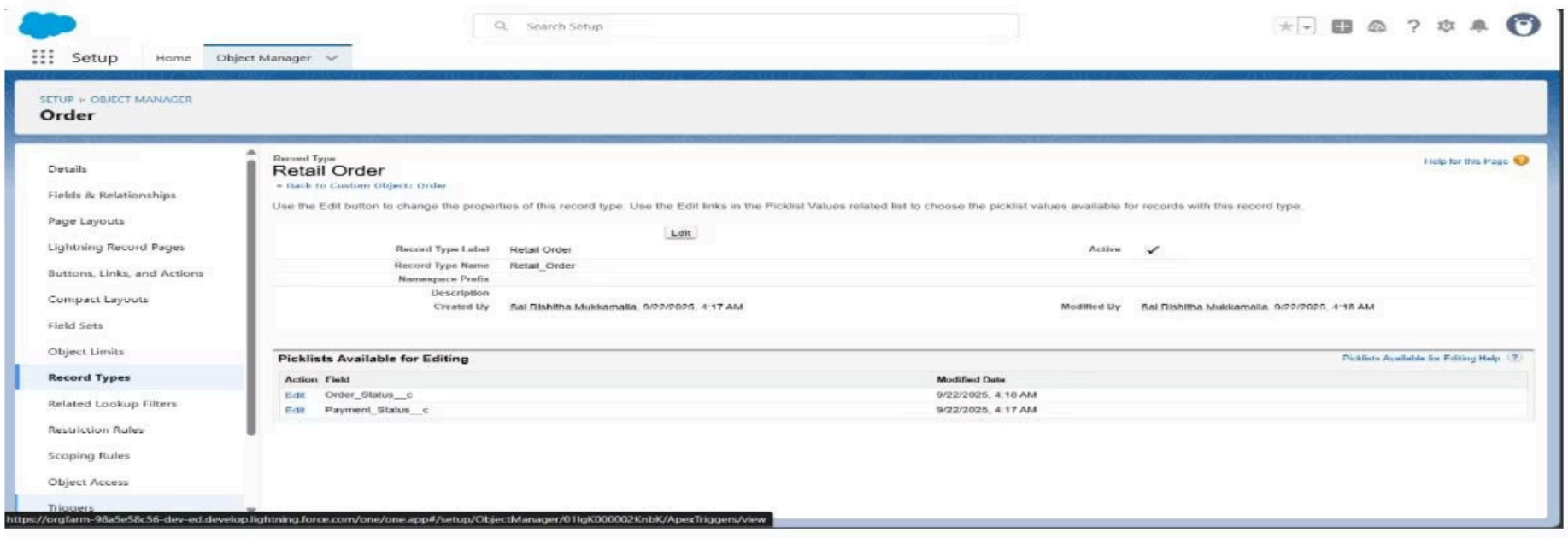
A **hierarchical relationship** is a special lookup available only for the **User** object, used to define a reporting structure between users.

Example in HotelConnect:

- Within the **User object**, hierarchical relationships were set up to represent reporting (e.g., Hotel Manager supervising Front Desk Staff and Housekeeping Staff).





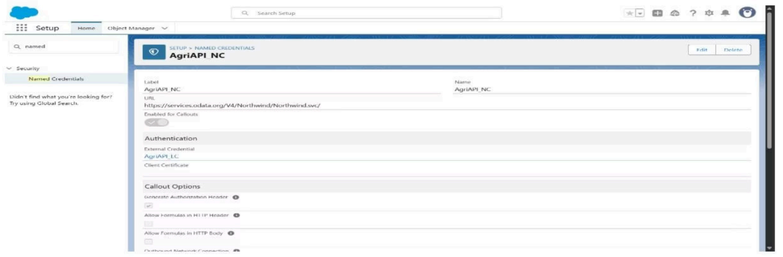


External Objects & System Integration

🔐 Named Credentials

Named Credentials are used to **securely store authentication details** (like usernames, passwords, OAuth tokens) for connecting Salesforce to an external system. Instead of hardcoding login info, Named Credentials centralize and manage them securely.

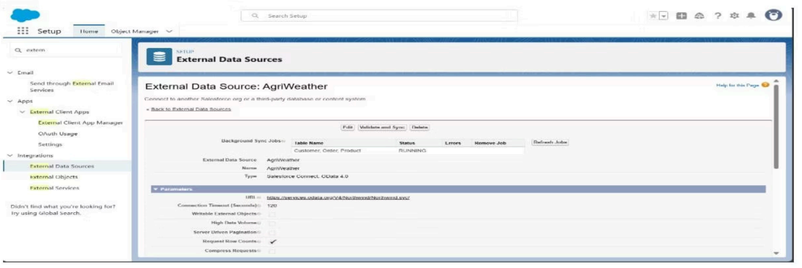
- In **HotelConnect**, a Named Credential was created for connecting to the **Hotel Management System / Property Management System (PMS) API**
- **Identity Type:** Named Principal (single login for all users)
- **Authentication Protocol:** OAuth 2.0 (with client credentials for secure API access)



🌐 External Data Source

An **External Data Source** defines the connection to an external system. This is where Salesforce knows **where the data is stored and how to access it**.

- **Data Source Type:** OData 4.0 (Hotel PMS / Booking Engine API)
- **Authentication:** Connected via the Named Credential
- After configuration, the **Validate and Sync** option was used to pull metadata from the external system

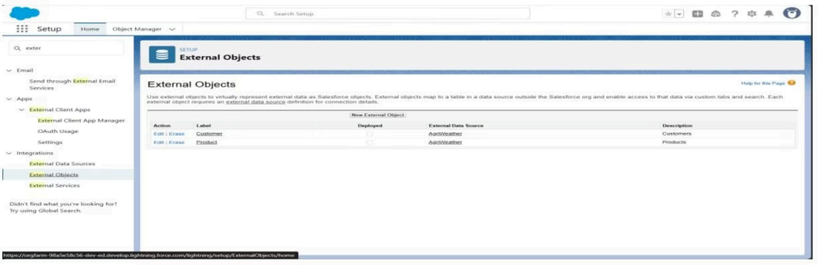


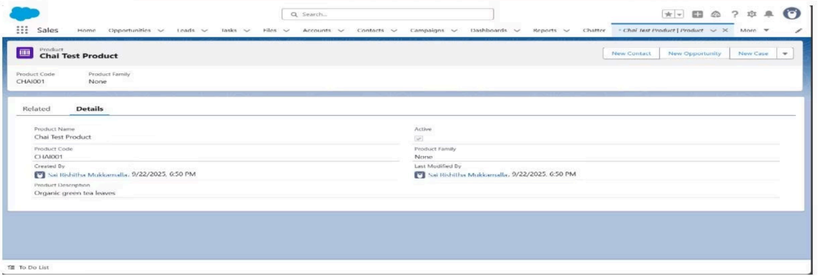
📦 External Objects

Once the external data source was validated and synced, Salesforce automatically created external objects:

- **HousekeepingSchedule__x**
- **RoomService__x**
- **MaintenanceRequest__x**
- **InventoryItem__x**

These behave like Salesforce objects but pull records directly from the external system in **real time**.





Conclusion

In Phase 3, I successfully designed and implemented the data model for the HotelConnect system by combining both Salesforce standard objects and custom objects that meet the project's requirements. Custom fields were created to capture hotel-specific data, and record types ensured that different booking types and processes are handled distinctly.

To improve usability, page layouts and compact layouts were customized, presenting information in a clear and meaningful way for hotel staff. The Schema Builder was used to visualize the overall data model, making it easier to understand relationships and dependencies.

Different types of relationships—Lookup, Master-Detail, and Hierarchical—were applied appropriately, ensuring proper data linkage, roll-up summaries, and ownership behaviors. For complex hotel operations, external objects were integrated through Named Credentials and External Data Sources, enabling Salesforce to display real-time data from property management systems.

Overall, Phase 3 provided a complete, scalable, and integrated data model for HotelConnect, ensuring that all future processes, automations, and analytics will be built on a strong and well-structured foundation for comprehensive hotel reservation and guest management.